

VŠB – TECHNICKÁ UNIVERZITA OSTRAVA

FAKULTA STROJNÍ

KATEDRA ROBOTIKY

BAKALÁŘSKÁ PRÁCE

VŠB – Technická univerzita Ostrava

Fakulta strojní

Katedra robotiky

Řízení motor driveru z Netduina

Motordriver Controlling from Netduino

Student:

Ing. Michal Ruttkay

Vedoucí diplomové práce:

Ing. Ján Babjak, Ph.D.

Ostrava 2018

Zadání bakalářské práce

Student: **Ing. Michal Ruttkay**
Studijní program: B2341 Strojírenství
Studijní obor: 2301R013 Robotika
Téma: **Řízení motor driveru z Netduina**
Motordriver Controlling from Netduino
Jazyk vypracování: čeština

Zásady pro vypracování:

1. Vypracujte rešerši motordriverů různých výrobců
2. Seznamte se s principy komunikace s těmito drivery přes CAN
3. Vytvořte knihovnu funkcí pro komunikaci z těmito drivery v prostředí Microframework/Keil
4. Na základě této knihovny vytvořte program demonstrující funkčnost
5. Práci doložte i v elektronické formě

Seznam doporučené odborné literatury:

1. Zásady pro vypracování diplomové (bakalářské) práce. [online] [cit. 2013-11-07]. Dostupné z [www: <http://robot.vsb.cz/pozadavky-na-prace>](http://robot.vsb.cz/pozadavky-na-prace)
2. ČSN 01 6910 *Úprava písemností psaných strojem nebo zpracovaných textovými editory*. Praha: Český normalizační institut, srpen 1997. 36 s.
3. ISO 690. *Bibliografické citace: Obsah, forma a struktura*. Praha: Český normalizační institut, 1996. 32 s.
4. Dokumentace ke zvolenému vývojovému prostředí
5. dokumentace k použitému mikroprocesoru
6. dokumentace k motor driverům EPOS či RoboteQ

Formální náležitosti a rozsah bakalářské práce stanoví pokyny pro vypracování zveřejněné na webových stránkách fakulty.

Vedoucí bakalářské práce: **Ing. Ján Babjak, Ph.D.**

Datum zadání: 08.12.2017

Datum odevzdání: 21.05.2018



prof. Dr. Ing. Petr Novák
vedoucí katedry



doc. Ing. Ivo Hlavatý, Ph.D.
děkan fakulty

Místopřísežné prohlášení studenta

Prohlašuji, že jsem celou diplomovou (bakalářskou) práci včetně příloh vypracoval samostatně pod vedením vedoucího diplomové (bakalářské) práce a uvedl jsem všechny použité podklady a literaturu.

V Ostravě

.....

podpis studenta

Prohlašuji, že

- jsem byl seznámen s tím, že na moji diplomovou (bakalářskou) práci se plně vztahuje zákon č. 121/2000 Sb., autorský zákon, zejména § 35 – užití díla v rámci občanských a náboženských obřadů, v rámci školních představení a užití díla školního a § 60 – školní dílo.
- beru na vědomí, že Vysoká škola báňská – Technická univerzita Ostrava (dále jen „VŠB-TUO“) má právo nevýdělečně ke své vnitřní potřebě diplomovou (bakalářskou) práci užít (§ 35 odst. 3).
- souhlasím s tím, že diplomová (bakalářská) práce bude v elektronické podobě uložena v Ústřední knihovně VŠB-TUO k nahlédnutí a jeden výtisk bude uložen u vedoucího diplomové (bakalářské) práce. Souhlasím s tím, že údaje o kvalifikační práci budou zveřejněny v informačním systému VŠB-TUO.
- bylo sjednáno, že s VŠB-TUO, v případě zájmu z její strany, uzavřu licenční smlouvu s oprávněním užít dílo v rozsahu § 12 odst. 4 autorského zákona.
- bylo sjednáno, že užít své dílo – diplomovou (bakalářskou) práci nebo poskytnout licenci k jejímu využití mohu jen se souhlasem VŠB-TUO, která je oprávněna v takovém případě ode mne požadovat přiměřený příspěvek na úhradu nákladů, které byly VŠB-TUO na vytvoření díla vynaloženy (až do jejich skutečné výše).
- beru na vědomí, že odevzdáním své práce souhlasím se zveřejněním své práce podle zákona č. 111/1998 Sb., o vysokých školách a o změně a doplnění dalších zákonů (zákon o vysokých školách), ve znění pozdějších předpisů, bez ohledu na výsledek její obhajoby.

V Ostravě :.....

.....

podpis

Jméno a příjmení autora práce:

Adresa trvalého pobytu autora práce:

ANOTACE DIPLOMOVÉ PRÁCE

Bakalářská práce se zabývá řízením motor driveru z mikrokontroléru přes sběrnici CAN. Zaměřuje se výhradně na motor driver EPOS2 50/5 firmy Maxon, pro kterou byly vytvořeny knihovny v jazyce C. Tyto knihovny byly implementovány k otestování funkčnosti motoru.

ANNOTATION OF MASTER THESIS

The Bachelor thesis deals with motor driver controlling from the microcontroller via the CAN bus. It focuses exclusively on the Maxon's motor driver EPOS2 50/5, for which libraries have been created in language C. These libraries have been implemented to test the motor's functionality.

Poděkování

Děkuji vedoucímu bakalářské práce Ing. Jánů Babjakovi, Ph.D. za odbornou pomoc, vynaloženou snahu o vysvětlení a za další cenné rady, které mi byly nápomocné při zpracování mé bakalářské práce.

Obsah

Seznam obrázků	9
Seznam tabulek	10
Seznam použitých značek a symbolů	11
Úvod	12
1. Motor driver vs Motor kontrolér (mikrokontrolér)	13
2. Motor driver	14
2.1. Maxon motor driver.....	14
2.1.1. Typy motor driveru Maxon	15
2.1.2. Komunikace motor driveru.....	16
2.1.3. Operační mody EPOS.....	17
2.2. RoboteQ motor driver.....	18
2.2.1. Operační mody RoboteQ	18
2.2.2. Typy motor driveru RoboteQ	20
3. Motor kontrolér (Mikrokontrolér)	21
3.1. Netduino	22
3.2. STMicroelectronics	23
4. Komunikace CANopen	24
4.1. Vlastnosti protokolu CAN	24
4.2. Typy zpráv	24
4.3. Popis CANopen	27
5. Stejnoseměrné motory	28
5.1. Motory DC kartáčové	28
5.2. Motory BLDC	29
6. Použitý hardware	31
6.1. Maxon motor EC 90 flat s MILE encoderem	31
6.2. Maxon motor driver EPOS2 50/5	32
6.2.1. Konektor k motoru	32
6.2.2. Konektor k Encoderu	33
6.2.3. Konektor k Hall senzoru	33
6.3. Mikrokontrolér	34
6.4. Schéma zapojení	35

7. Program mikrokontroléru.....	36
7.1. Vytvoření projektu v STM32CubeMX	36
7.2. Vývojové prostředí KEIL	37
7.3. Knihovna funkcí	37
7.4. Schéma komunikace s motor driverem	38
7.5. Vývojový diagram programu	39
Závěr	41
Seznam použité literatury	42

Seznam obrázků

Obrázek 1: Zjednodušená schématická struktura	13
Obrázek 2: Motor drivery firmy Maxon [4]	15
Obrázek 3: Konektor RS232 [3]	16
Obrázek 4: Konektor CANbus.....	16
Obrázek 5: Epos - operační mody [5]	17
Obrázek 6: Motor driver firmy RoboteQ	18
Obrázek 7: Motor kontrolér	21
Obrázek 8: Netduino N3 WiFi [12]	22
Obrázek 9: Mikrokontrolér	23
Obrázek 10: Datový rámec CAN 2.0A	25
Obrázek 11: Rozšířený datový rámec CAN 2.0B.....	25
Obrázek 12: Třívrstvý datový model	27
Obrázek 13: Schématické složení kartáčového motoru, upraveno z [22].....	29
Obrázek 14: Kartáčový motor [18].....	29
Obrázek 15: Schématické složení bezkartáčového motoru, upraveno z [22]	30
Obrázek 16: Bezkartáčový motor [18].....	30
Obrázek 17: Maxon motor EC 90 flat s MILE encoderem [24]	31
Obrázek 18: Motor driver EPOS2 50/5 [23].....	32
Obrázek 19: Propojky (JP2/JP3) – otevřeno (Výchozí stav) / zavřeno (Vpravo) [23] ...	32
Obrázek 20: Vývojová deska STM32F103ZE [21]	34
Obrázek 21: Schéma zapojení.....	35
Obrázek 22: STM32CubeMX nastavení mikrokontroléru	36
Obrázek 23: Keil nastavení debuggeru	37
Obrázek 24: Schéma při komunikaci s EPOS2 50/5	38
Obrázek 25: Vývojový diagram programu	39

Seznam tabulek

Tabulka 1: Technické parametry motor driverů rodiny Maxon [4].....	15
Tabulka 2: RoboteQ- typy bezkartáčových motorů [10]	20
Tabulka 3: RoboteQ- typy kartáčových motorů [9].....	20
Tabulka 4: Netduino 3 - Druhy modelů [12]	22
Tabulka 5: Porovnání kartáčových a bezkartáčových motorů [16]	28
Tabulka 6: Zapojení motoru [23].....	33
Tabulka 7: Zapojení encoderu [23].....	33
Tabulka 8: Zapojení Hall senzoru [23]	33

Seznam použitých značek a symbolů

V	Volt , základní jednotka napětí v SI soustavě
A	Ampér , odvozená jednotka proudu v SI soustavě
W	Watt , odvozená jednotka proudu v SI soustavě
EPOS	Easy to use POS ition S ystem
PWM	P ulse W idth M odulation
CAN	C ontroller A rea N etwork
MCU	Mikrokontrolér
ADC	A nalog to D igital C onverter
RCC	R eset and C lock C ontrol
IDE	I ntegrated D evelopment E nvironment
HAL	H ardware A bstraction L ayer

Úvod

Provoz složitých, ale i jednoduchých zařízení vyžaduje regulaci otáček, momentu, výkonu nebo záběrového proudu motoru. Rozvojem elektrického ovládání, automatizace systému a zařízení, zejména v oblasti výkonových mikroprocesorů se rozšířilo programové řízení motorů využívajících pulsního řízení založené na pulsně-šířkové modulaci (PWM).

Komunikační CAN protokol v dnešní době používá stále více výrobců v zařízeních v oblasti automatizace, řízení, robotiky a v elektronické výbavě automobilů.

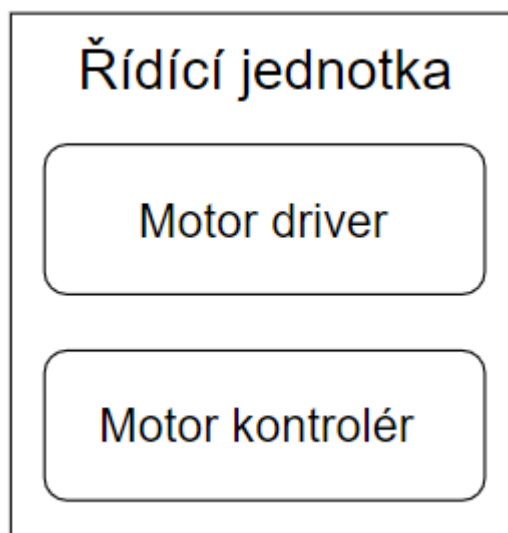
Cílem bakalářské práce je řízení motor driveru pomocí mikrokontroléru přes sběrnici CAN. Motor driver poskytuje motoru dostatečné napětí a proud, který je nutný k řízení motoru. Práce popisuje komunikační rozhraní CAN, dvě rodiny motor driverů od společností Maxon a Roboteq a také mikrokontrolery STMicroelectronics a Netduino.

1. Motor driver vs Motor kontrolér (mikrokontrolér)

K řízení motorizované součásti je zapotřebí dostatečné množství proudu, který standartní čip mikrokontroléru není schopen poskytnout ke spolehlivému řízení malého motoru, natož většího. Vzhledem k tomu, že mikrokontrolér není schopen řídit motor, je zapotřebí zařízení, které to dokáže, což je Motor driver.

Motor driver používá větší čip nebo diskrétní tranzistory FET, které jsou schopné poskytnout větší množství proudu a napětí než standartních 5V / 3,3V z mikrokontroléru. Takhle se ovládají velké zátěže, malým signálem z mikrokontroléru.

K řízení motoru jsou zapotřebí oba zmíněné komponenty, které jsou důležité k ovládání motoru. Motor driver se stará o výkonovou část, zatím co logika a digitální řízení je prováděno motor kontrolérem. [1]



Obrázek 1: Zjednodušená schématická struktura
řídící jednotky

2. Motor driver

Motor driver je proudový a napěťový zesilovač, jehož funkcí je přijmout řídicí signál o nízké úrovni a ten poté zesílit na požadované napětí a proud, které jsou nutné k řízení motoru. Součástí používaných motor driverů od firem Maxon a RoboteQ je motor kontrolér, který dokáže spolehlivě řídit motor.

Existuje mnoho druhů motor driveru, které jsou kategorizovány např. maximálním napájecím napětím, maximálním výstupním proudem, jmenovitým výkonem, zatížením a počtem vstupů. Nejběžnější hodnoty maximálního napájecího napětí jsou 36V a 52V s počtem pinů 1-4.

Motor drivery se používají v řadě aplikací [2] :

- Krokový motor
- LED a žárovkové displeje
- Automobilové aplikace
- Audiovizuální zařízení
- PC periferní zařízení
- Audiosystémy automobilů
- Automobilové navigační systémy

2.1. Maxon motor driver

Švýcarská firma Maxon motor vyvinula univerzální řídicí jednotku k řízení stejnosměrných motorů, které jsou určeny pro řízení a regulaci proudu tekoucího do vinutí, rychlosti a polohy hřídele komutátorových a elektronicky komutovaných strojů (EC motory). Řídicí jednotky polohy firmy Maxon se nazývají EPOS – z počátečních anglických písmen „Easy to use POSition System“ v překladu jednoduše použitelný systém pro řízení polohy. Tyto jednotky jsou zcela digitální a starají se o přesné a rychlé řízení motorů podle příkazů z mikrokontroléru nebo PC. Základní funkce jednotky EPOS jsou v okamžité odezvě na aktuální údaje podle požadovaných parametrů. Až poté jsou data odeslány nadřazenému systému, který stanoví další průběh. Jednotka by tedy měla motoru přidávat (odebírat) potřebné množství proudu v závislosti na zatížení tak, aby byly otáčky motoru konstantní. K ovládání mohou být použity kartáčové DC motory s inkrementálním snímačem, tak i bezkartáčové motory s hallovými sondami a inkrementálním snímačem. EPOS podporuje i EC motory pro řízení s menším rozlišením. [3]



Obrázek 2: Motor drivery firmy Maxon [4]

2.1.1. Typy motor driveru Maxon

Firma Maxon vyrábí řídicí jednotky typu EPOS v pěti výkonových variantách, které se v názvu liší označením. První dvojčíslí označuje napájecí napětí a číslice za lomítkem udávají maximální proud, který je schopna dodat.[3]

Tabulka 1: Technické parametry motor driverů rodiny Maxon [4]

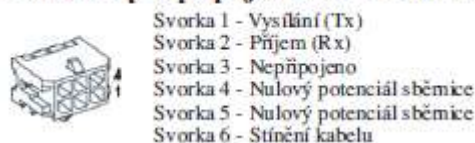
Parametry	EPOS2 24/2	EPOS2 Module 36/2	EPOS2 24/5	EPOS2 50/5	EPOS2 70/10
Max. výkon DC a EC motoru	48W	72	120	250	700
Ovládání PTP	1 osa	1 osa	1 osa	1 osa	1 osa
Mod interpolované polohy	PVT	PVT	PVT	PVT	PVT
CANOpen	Ano	Ano	Ano	Ano	Ano
Digitální vstupy	6	3	6	11	10
Digitální výstupy	2	3	4	5	5
Analogové vstupy	2	3	2	2	2
Analogové výstupy	-	-	-	1	-
Design	Miniaturní	Miniaturní deska s otevřenou elektronikou	Kompaktní	Kompaktní	Robustní

2.1.2. Komunikace motor driveru

Sériová komunikace přes RS232

Princip sériové komunikace slouží pro komunikaci POINT-to-POINT, která je založena na standartu EIA-RS232. V principu jde o vysílání dat z MASTER – PC do SLAVE – EPOS, přičemž jde komunikovat pouze s jednou řídicí jednotkou. Nejčastěji se používá negativní bipolární logika, kde logická „1“ představuje záporné napětí v rozmezí -3 až -25V a logická „0“ kladné napětí v rozsahu 3V až 25V. Komunikace mezi zařízeními je zajištěna třemi vodiči – krouceným párem vodičů a ochranným vodičem. [5]

Konektor pro připojení sběrnice RS 232

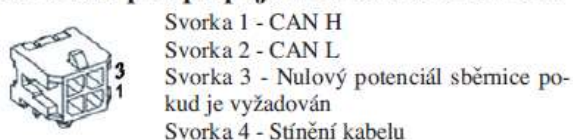


Obrázek 3: Konektor RS232 [3]

Komunikace po sběrnici CANBus

Řídicí jednotky EPOS obsahují konektor komunikačního rozhraní standartu CANBus, který podporuje až 127 jednotek. Komunikace mezi jednotkami využívá znalosti unikátního identifikačního čísla (ID). ID číslo se přiřazuje pomocí přepínače poblíž konektoru CANbus. Identifikace stavu jednotky je reprezentována dvojbarevnou LED diodou (Červená – porucha, Blikání zelené – neaktivní stav jednotky, Zelená – pracuje v operačním modu).

Konektor pro připojení sběrnice CANBus

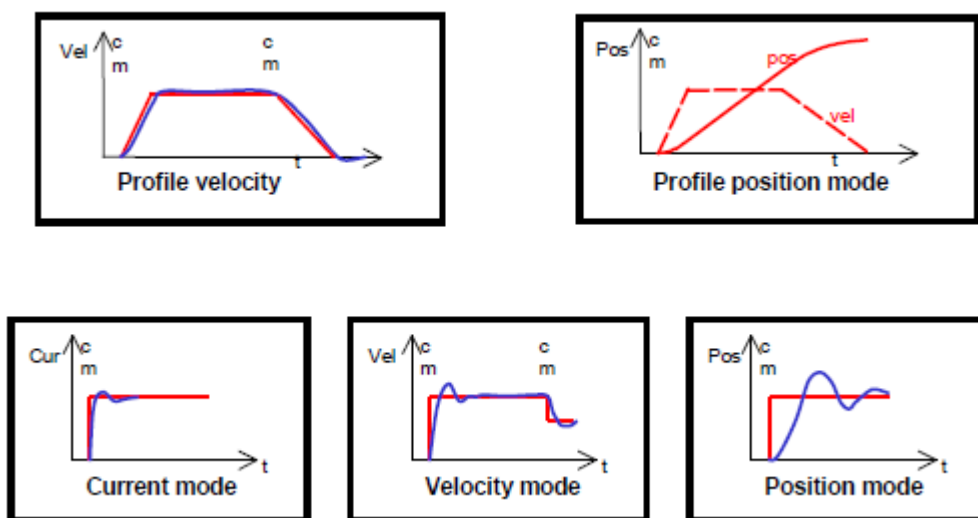


Obrázek 4: Konektor CANbus

2.1.3. Operační módy EPOS

Chování zařízení závisí na aktivovaném provozním režimu. V řídicích jednotkách EPOS lze nastavit tyto uvedené módy [6]:

- **Homing mode** – obsahuje metody k nalezení referenční pozice
- **Profile position mode** – slouží k polohování a může omezit rychlost a zrychlení
- **Position mode** – pozice je zadána přímo hodnotou
- **Profile velocity mode** – slouží k řízení rychlosti pohonu bez speciálních požadavků na pohon
- **Velocity mode** – rychlost je zadána přímo hodnotou
- **Current mode** – aktivovaná je pouze proudová smyčka k omezení rychlostí
- **Diagnostic mode** – diagnostika v grafickém rozhraní
- **Master encoder mode** – pozice je definována externím inkrementálním čidlem
- **Step/Direction mode** – chová se jako servopohon krokového motoru



Obrázek 5: Epos - operační módy [5]

2.2. RoboteQ motor driver

Motor drivers firmy RoboteQ poskytuje uživatelské rozhraní RoboRealm k řízení motorů, především od značek RoboteQ. Obsahují vysoce výkonné mikrokontroléry a vstupy kvantového kodéru pro řízení pohybu pomocí pokročilých algoritmů. Také jsou vybaveny několika analogovými, pulzními a digitálními vstupy / výstupy, které lze přenastavit např. na koncové spínače a zpětné vazby. [7]

Dvoukanálové RoboteQ motor drivers obsahují regulátory, které mohou být ovládané nezávisle nebo hybridně tak, aby bylo možné např. nastavit směr otáčení vozidla koordinaci pohybu každého motoru. Pro spolehlivý a bezpečný provoz jsou do regulátoru umístěny bezpečnostní prvky. Ovládání lze automatizovat a přizpůsobovat pomocí základních jazykových skriptů. [8]



Obrázek 6: Motor driver firmy RoboteQ

2.2.1. Operační mody RoboteQ

Multiple Motion Modes

Každý kanál mikrokontroléru lze nastavit do jednotlivých módů [25]:

- **Open Loop Speed mode** – regulátor poskytuje takové množství energie, které je potřebné ke konstantním otáčkám motoru.
- **Closed Loop Speed modes**
 - **Closed Loop Speed mode** – tento režim se používá k měření skutečné rychlosti motoru optickým snímačem nebo analogovým otáčkoměrem, v případě bezkartáčových motorů pomocí Hallových sond. V případě změny rychlosti

v důsledku zatížení dochází k automatickému navýšení výkonu. Tento režim se využívá k přesnému řízení motoru.

- **Closed Loop Speed Position** – v tomto režimu regulátor vypočítává polohu, ve které musí být každou 1ms. Poté PID regulátor porovnává aktuální polohu s vypočtenou a použije potřebný výkon k dosažení požadované polohy. Používá se pro přesné řízení při velmi pomalých rychlostech.
- **Closed Loop Position modes** – v tomto režimu je k motoru připojen potenciometr nebo snímač, který se používá k porovnání úhlové polohy s požadovanou polohou.

Tři různé režimy polohy:

- **Position Relative** – motor se pohybuje z aktuální polohy do cíle pomocí řízeného zpomalení, zrychlení a stále rychlostí. K zjištění aktuální polohy se používají Hallovy a analogové snímače.
- **Count Position** – jedná se o podobný režim jako Position Relative, ale k zjištění pozice se využívají všechny snímače se zpětnou vazbou. Poloha motoru může být změněna na libovolnou polohu pomocí řízeného profilu zrychlení a rychlosti.
- **Position Tracking** – v tomto případě se jedná o podobný režim jako Position Relative, ale pouze s PID regulátorem. Motor se nastaví do požadované polohy co nejrychleji.
- **Torque mode** – regulátor nastaví a udržuje nastavenou hodnotu proudu.

Mixed Motor Command Modes

Dva kanály mohou být připojeny ke dvěma motorům, které reagují na příkazy přijaté ze dvou vstupních kanálů. Pomocí dvoukanálových vstupů lze motory ovládat samostatně nebo kombinovaným způsobem.

- **Independent Speed Command** – režim, kde každý motor je ovládán nezávisle. Vhodné pro obecné řízení motorů.
- **Mixed Speed/Steering Command** – hybridní režim umožňující motorům pracovat v kombinaci s pohybem a otáčením vozidla. Tento režim provozu zajišťuje efektivní metodu pohybu.

2.2.2. Typy motor driveru RoboteQ

Motor drivery firmy RoboteQ, lze rozdělit podle typu motoru, který bude řídit:

Bezkartáčové motory

Tabulka 2: RoboteQ- typy bezkartáčových motorů [10]

Označení	Popis
SBL13xx/23 xx Family	Regulátor stejnosměrných motorů. Nízký výkon a kompaktní rozměry. 1/2 kanálové zařízení. Využití pro řízení strojů a robotiku. Řídící jednotky SBL1xxx podporují lichoběžníkový režim a vyžadují pouze Hallovy snímače pro pohon motorů. SBL2xxx podporuje také sinusový režim.
MBL1xxx Family	Regulátory stejnosměrných motorů. Středně výkonové jednobukálové regulátory. Využití pro řízení strojů a robotiku. K dispozici ve verzích "Basic" a "Advanced Control". Regulátory základních verzí pracují pouze v trapézovém režimu a vyžadují Hallovy snímače pro pohon motoru. Pokročilé modely řízení podporují sinusový režim.
FBL23xx Family	Regulátory stejnosměrných motorů. Vysoko výkonové dvoukanálové motory. Využití pro AGV a malé elektrické vozy. Podpora trapézové komutace a sinusový režim.
HBLxxxx Family	Regulátory stejnosměrných motorů. Vysoce výkonové jednobukálové nebo středně výkonové dvoukanálové motory. Využití pro mobilní roboty a malé elektrické vozy. K dispozici ve verzích "Basic" a "Advanced Control". Regulátory základních verzí pracují pouze v trapézovém režimu a vyžadují Hallovy snímače pro pohon motorů. Pokročilé modely řízení podporují sinusový režim.
GBL26xx Family	Regulátory stejnosměrných motorů. Vysoce výkonové jednobukálové a dvoukanálové motory. Volitelné chlazení vodou. Využití pro manipulaci s těžkými materiály, automatizovaně řízené vozy, zemědělské roboty a další aplikace s vysokým výkonem. Podporuje trapézovou komutaci a sinusový režim.

Kartáčové motory

Tabulka 3: RoboteQ- typy kartáčových motorů [9]

Označení	Popis
SDC21xx Family	Regulátor stejnosměrných motorů. Nízko výkonové jedno a dvoukanálové regulátory. Využití pro robotiku, řízení strojů a automatizaci.
MDCxxxx Family	Regulátory stejnosměrných motorů. Středně výkonové jedno nebo dvoukanálové regulátory. Využití pro robotiku, řízení strojů a automatizaci.
FDC32xx Family	Regulátory stejnosměrných motorů. Středně výkonové tříkanálové regulátory. Využití pro simulaci pohybu, podvodní roboty a průmyslovou automatizaci.
HDC/XDC24 xx Family	Regulátory stejnosměrných motorů. Vysoce výkonové dvoukanálové regulátory. K dispozici konfigurace s jedním kanálem až do 300A. Využití pro vysokovýkonové mobilní roboty a malé elektrické vozy.
RGDC1xxx Family	Regulátory stejnosměrných motorů. Vysoce výkonové jednobukálové regulátory, proud až 300A. Několik možností napětí až do 100V. Využití pro elektrická vozidla, golfová vozidla, manipulační zařízení, elektrické čluny, zemědělské roboty a pro další aplikace s vysokým výkonem.

3. Motor kontrolér (Mikrokontrolér)

Mikrokontrolér je samostatným systémem s periferními zařízeními, pamětí a procesorem. Lze ho použít jako integrovaný systém (embedded system). V dnešní době je většina programovatelných mikrokontrolérů uložena v elektronických přístrojích nebo strojích, proto se také uvádí pod cizím názvem embedded systém. Některé mikrokontroléry jsou sofistikovanější a jiné mají minimální požadavky na paměť s nízkou složitostí softwaru. Vstupní a výstupní zařízení zahrnují mimo jiné LCD displeje, relé, spínače a snímače pro vlhkost, teplotu nebo intenzitu osvětlení. [11]

Mikrokontroléry se dají kategorizovat podle:

- **Počet bitů** – kolik bitů je schopný zpracovat během jedné operace
- **I/O porty** – vstupní/výstupní rozhraní
- **Velikost paměti RAM** – volně přístupná paměť na provozní data (po vypnutí se smaže)
- **Velikost paměti ROM** – paměť slouží k nahrávání vlastního programu, zachová se i po vypnutí
- **Napájecí napětí** – potřebné napětí mikrokontroléru (3,3V / 5V)
- **Frekvence procesoru** – počet operací, který je schopen vykonat během jedné sekundy



Obrázek 7: Motor kontrolér

3.1. Netduino

Netduino je open source hardwarová platforma pro aplikace využívající .Net MicroFramework, která je součástí Microsoft Visual Studia. Koncept je podobný platformě Arduino, ale je výkonnější a namísto psaní aplikací v C/C++ jsou aplikace napsány v jazyce C#, který přináší výkonnější jazykové konstrukce např. manipulace s událostmi, vlákny. Netduino se používá k vytváření gadgetů pro domácí automatizaci, průmyslové stroje např. CNC stroje a další.

Jednotlivé modely Netduina nemají integrované CAN rozhraní pro zmíněné řízení motor driveru a proto je nutné vybrat adekvátní náhradu.

V současnosti nejvyšším modelem je Netduino 3, který se nabízí ve třech verzích lišících se síťovou vybaveností a velikostí paměti Flash viz Tabulka 4. [12]

Tabulka 4: Netduino 3 - Druhy modelů [12]

Model	MCU	Flash	RAM	Network
N3	Cortex-M4 @ 168MHz	384KB	164+ KB	n/a
N3 Ethernet	Cortex-M4 @ 168MHz	1408KB	164+ KB	10/100Mbs Ethernet
N3 WiFi	Cortex-M4 @ 168MHz	1408KB	164+ KB	802.11b/g/n with SSL/TLS 1.2 Sup



Obrázek 8: Netduino N3 WiFi [12]

3.2. STMicroelectronics

Společnost STMicroelectronics nabízí spoustu produktů v oblasti mikroprocesorů (MCU) od robustních 8-bitových MCU, až po 32-bitové Cortex MCU. Mikrokontroléry s označením STM32 představují úspěšnou řadu 32 bitových MCU s jádrem ARM Cortex M. Architektura zmíněných MCU je typu RISC (redukovaná instrukční sada), což označuje malý soubor instrukcí o stejné délce 32 bitů. Mikrokontroléry STM32 s jádrem Cortex M0, M0+ a M1 jsou postaveny na Von Neumannově paměťové architektuře, která využívá sběrnici pro data i instrukce. Vyšší řady mají samostatnou sběrnici jak pro data, tak i pro instrukce, což je typické pro Harwardskou architekturu.

V současné době série STM32 obsahuje deset řad mikrokontrolérů, kde každá řada je založena na zmíněném MCU procesoru (Cortex-M7F , Cortex-M4F , Cortex-M3 , Cortex-M0 + nebo Cortex-M0 ARM).

Pro usnadnění programování byl vytvořen grafický průvodce (STM32CubeMX) generující inicializační kód v jazyce C. Obsahuje také hardwarovou abstrakční vrstvu STM32Cube(HAL). Ovladače HAL obsahují kompletní sadu API připravených k použití, které zjednodušují implementaci uživatelských aplikací.[13]



Obrázek 9: Mikrokontrolér

4. Komunikace CANopen

CANopen patří mezi vyšší komunikační protokoly pro vestavené sítě a je vytvořena na základě sběrnice CAN (Controller Area Network). Jeho standardní protokol je široce konfigurovatelný pro vestavné řídicí sítě a zařízení s řízenými pohyblivými částmi (manipulátory). V současné době již přední výrobci implementovali podporu protokolu do svých produktů, proto jsou využívány v různých odvětvích průmyslu (lékařský, automobilový, námořní systémy a automatizace ve stavebnictví) především kvůli nízké ceně, spolehlivosti, vysoké přenosové rychlosti, snadné rozšíření a dostupnosti.[14]

4.1. Vlastnosti protokolu CAN

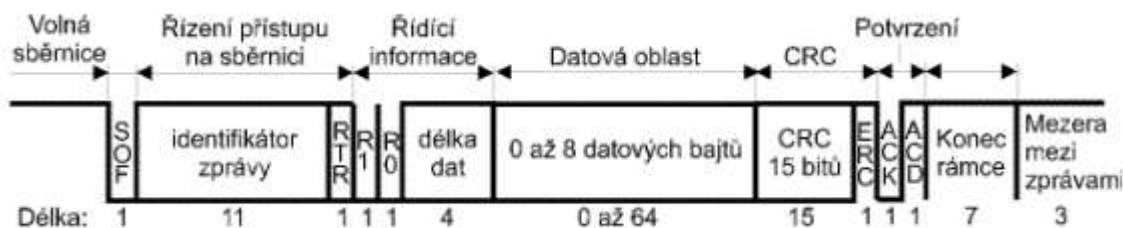
CAN vznikl k řízení distribuovaných systémů v reálném čase s přenosovou rychlostí do 1Mbit/s a s vysokým stupněm zabezpečení přenosu proti chybám. Využívá komunikace typu multi-master, kde více jak jedno zařízení může přebrat řízení sběrnice a zároveň k ní může být připojeno. Sběrnici v tomto případě nemusíme řídit z nadřazeného místa, což přináší zjednodušení a spolehlivost v případě výpadku. Komunikace po sběrnici mezi dvěma uzly probíhá pomocí zpráv (datová zpráva a žádost o data) a managementem sítě (signalizace chyb, pozastavení komunikace), je obstarána pomocí speciálních zpráv (chybové zprávy a zprávy o přetížení). Odeslané zprávy po sběrnici neobsahují žádnou informaci o svém cíli a jsou přijaty všemi zařízeními připojenými ke sběrnici. Zprávy obsahují identifikátor významu přenášené zprávy a její prioritu (Identifikátor 0 – nejvyšší priorita), která zaručuje v případě kolize přednostní doručení. Identifikátor zajistí přijmutí zprávy, která se ho týká.

4.2. Typy zpráv

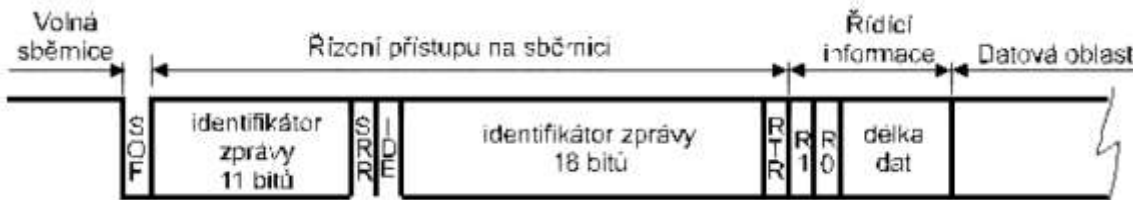
CAN protokol využívá dvě zprávy k datové komunikaci a dvě zprávy k managementu komunikace. Mezi zprávy datové komunikace patří datová zpráva, která je základním prvkem komunikace po sběrnici a umožňuje vyslat na sběrnici 0-8 datových bajtů. Druhá zpráva datové komunikace je zpráva na vyžádání dat, kdy zařízení žádá ostatní zařízení na sběrnici o zaslání požadovaných dat. Zprávy managementu komunikace (chybová zpráva a zpráva o přetížení) slouží k signalizaci detekovaných chyb, eliminaci chybných zpráv a vyžádání prodlevy v komunikaci.

Datová zpráva (Data Frame)

Existují dva typy datových zpráv, které jsou definovány různými specifikacemi. Specifikace 2.0A je v literatuře označována jako standardní formát zprávy, přičemž specifikace 2.0B je definována jako rozšířený formát vyznačující se delším identifikátorem zprávy (11 bitů – standardní formát, 29 bitů – rozšířený formát). Vysílání datové zprávy se uskuteční pouze, je-li sběrnice volná.



Obrázek 10: Datový rámec CAN 2.0A



Obrázek 11: Rozšířený datový rámec CAN 2.0B

Význam jednotlivých částí datové zprávy podle specifikace 2.0A

- Začátek zprávy (SOF = Start Of Frame)
 - 1 bit dominant
- Řízení přístupu na sběrnici – určení priority zprávy
 - Identifikátor zprávy – 11 bitů, udává význam přenášené zprávy
 - RTR bit (Remote Request) – 1 bit, příznak udává, zda se jedná o datovou zprávu nebo o žádost o vyslání dat. V datové zprávě musí být tento bit dominant, v žádosti o data recessive.
- Řídicí informace (Control Field)
 - R0, R1 – rezervované bity
 - Délka dat – 4 bity, počet přenášených datových bajtů ve zprávě. Povolené hodnoty jsou 0 až 8.
- Datová oblast (Data Field) – datové bajty zprávy
 - Maximálně 8 bajtů je vysláno od MSB CRC (CRC Field) – 16 bitů, zabezpečovací CRC kód
- CRC kód – 15 bitů
- ERC(CRC oddělovač)
 - 1 bit recessive
- Potvrzení (ACK Field)
 - 2 bity

- ACK (bit potvrzení)
 - 1 bit
- ACD (oddělovač potvrzení)
 - 1 bit recessive
- Konec zprávy (End Of Frame)
 - 7 bitů recessive
- Mezera mezi zprávami (Interframe Space) – odděluje dvě zprávy
 - 3 bity recessive

Žádost o data (Remote Frame)

Zpráva žádostí o data se liší pouze bitem RTR (řízení přístupu na sběrnici), který je nastaven na úroveň recessive a chybí zde datová oblast. Pokud zařízení žádá o zaslání dat, nastaví identifikátor zprávy na hodnotu odpovídající hodnotě zprávy, jejíž zaslání požaduje. Pokud ve stejném čase žádá o zaslání dat a jiné data se stejným identifikátorem vysílá, přednost v přístupu na sběrnici získá zařízení vysílající datovou zprávu díky RTR bitu v dominantní hodnotě (vyšší priorita).

Chybová zpráva (Error Frame)

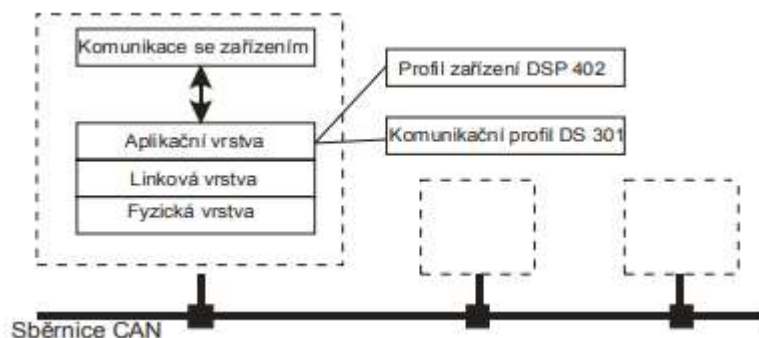
Tato zpráva slouží k signalizaci chyb vzniklých na sběrnici CAN. V případě detekce chyby (chyba bitu, chyba CRC, chyba vkládání bitů, chyba rámce) v přenášené zprávě dochází k okamžitému vygenerování chybového rámce na sběrnici. Podle stavu, ve kterém zařízení eviduje chybu na sběrnici, dochází k vygenerování příznaku složeného ze šesti bitů. Příznak může být buď aktivní (šest bitů dominant) nebo pasivní (šest bitů recessive). Aktivní příznak je vytvořen v případě poškozené zprávy (pravidla vkládání bitu), tuto zprávu začnou posílat i zařízení, která jsou připojena na sběrnici. Po odeslání chybového příznaku dochází na zařízeních k vysílání bitů recessive na sběrnici a detekuje se stav sběrnice. Detekuje-li bit ve stavu recessive, vysílá dalších 7 bitů recessive, které plní funkci oddělovače chyb. Hlášení chyb je signalizováno superpozicí všech chybových příznaků s délkou 6 až 12 bitů.

Zpráva o přetížení (Overload Frame)

Tento typ zprávy je podobný chybové zprávě, ale na rozdíl od ní umožňuje oddálení vyslání další datové zprávy nebo žádostí o data. Převážně se využívá v případech, kdy zařízení není schopno přijímat nebo zpracovávat zprávy v důsledku jeho vyčerpání.

4.3. Popis CANopen

Jedná se o síťový protokol, který využívá komunikační protokol CAN a odpovídající hardware s normou ISO 11898. Pro datové služby využívá třívrstvý datový model viz Obrázek 12.



Obrázek 12: Třívrstvý datový model

Fyzická vrstva – CAN Physical Layer

Fyzická vrstva popisuje elektrické specifikace (rozhraní mezi fyzickou vrstvou a fyzickými prostředky) a mechanické specifikace (délka a typy propojovacích vodičů a konektorů) sběrnice CAN a také fyzický přenos bitů po metalickém vedení.

Linková vrstva – CAN Data Link Layer

Je složena ze dvou podvrstev MAC (Medium Access Control) a LLC (Logical Link Control). Podvrstva MAC řídí přístup všech zařízení k médiu s ohledem na prioritu jednotlivých zpráv a jejím úkolem je kódování dat, detekce chyb, vkládání doplňkových bitů do komunikace a potvrzení správně přijatých zpráv. Druhá podvrstva zajišťuje filtraci přijatých zpráv.

Aplikační vrstva – CAN Application Layer

Aplikační vrstva umožňuje aplikačním procesům přistupovat ke komunikačnímu systému a dále provádí základní funkce (vytváření/udržení/rušení spojení, čtení/zápis přenášených dat).

5. Stejnosměrné motory

Téměř všechny elektromechanické pohyby, které vidíme kolem nás, jsou způsobeny buď střídavým, nebo stejnosměrným motorem. V této kapitole jsou popsány stejnosměrné kartáčové a bezkartáčové motory, které převádí stejnosměrnou elektrickou energii na energii mechanickou. V následující tabulce jsou uvedeny výhody jednotlivých motorů.

Tabulka 5: Porovnání kartáčových a bezkartáčových motorů [16]

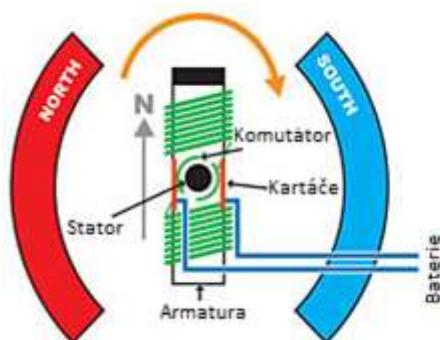
Výhody motoru	
DC motory	BLDC motory
Nízké náklady na konstrukci	Nižší náklady na údržbu
Měnitelné součásti k prodloužení životnosti	Efektivnost při všech rychlostech
Jednoduchý a levný přístroj	Vysoká účinnost a vysoký poměr výkon / rozměr
Nepotřebuje kontrolér pro stálou rychlost	Snížená velikost s lepšími tepelnými vlastnostmi
Vhodné do extrémního prostředí	Větší rozsah otáček a nižší šum

5.1. Motory DC kartáčové

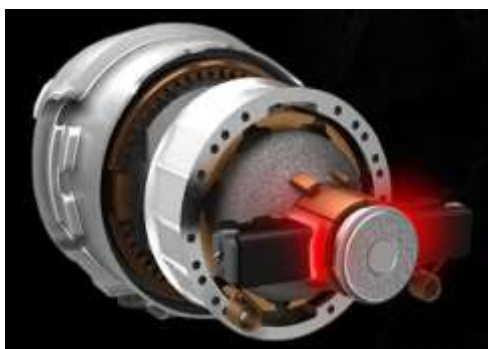
DC kartáčové motory jsou snadno ovladatelné a dostupné ve všech velikostech a tvarech. Skládají se ze základních komponentů: stator, rotor, kartáče a komutátor.

Stator vytváří kolem rotoru magnetické pole, které je tvořeno permanentními magnety nebo elektromagnetickým vinutím. Rotor neboli armatura používá sestavu navinutých drátů, přičemž působí jako dvoupólový elektromagnet. Směr proudu je obrácen dvakrát za jediný cyklus komutátorem (mechanickým spínačem), tím je usnadněn průchod proudu přes rotor.

Magnetické póly rotoru jsou přitahovány póly generované statorem, což způsobí otáčení rotoru. Komutace zajišťuje přepínání magnetických polí tak, aby byly neustále napájeny v jiných sekvencích. Při běhu motoru uhlíkové kartáče procházejí komutátorem a přicházejí do styku s různými segmenty, které jsou připevněny k rotoru. Tímto se vytváří dynamické magnetické pole, jestliže prochází napětí přes kartáče motoru. [17]



Obrázek 13: Schématické složení kartáčového motoru, upraveno z [22]



Obrázek 14: Kartáčový motor [18]

5.2. Motory BLDC

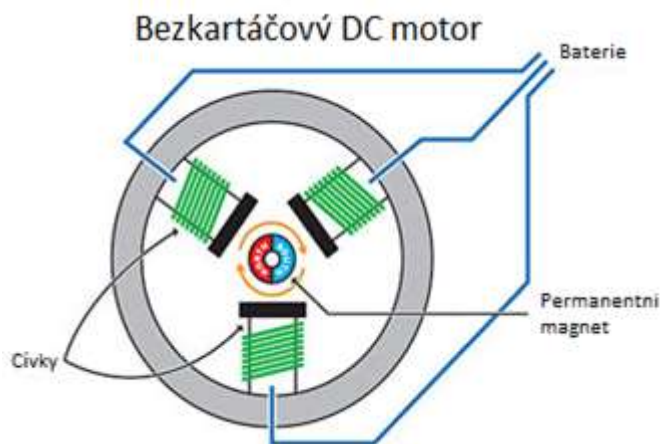
Stejnoseměrné bezkartáčové motory jsou typem synchronních motorů. To znamená, že magnetické pole generované statorem a rotorem se otáčí ve stejné frekvenci. BLDC motory se dodávají v jednofázových, dvoufázových a třífázových konfiguracích – nejrozšířenější.

Většina BLDC motorů má tři statorové vinutí spojené do hvězdy. Každé vinutí se skládá z několika cívek. Jedna nebo více cívek jsou umístěny do slotu, ve kterém jsou vzájemně propojeny tak, aby vytvářely vinutí. Toto vinutí je umístěno na každý obvod statoru, aby vytvořilo sudé počty pólů.

Rotor je vyroben z permanentního magnetu a obvykle obsahuje dva až osm párů pólů, které jsou umístěny střídavě. Na základě požadované hustoty magnetického pole v rotoru se zvolí vhodný magnetický materiál, nejčastěji feritové magnety.

Komutace BLDC motoru je řízená elektronicky na rozdíl od kartáčových motorů. Ve statoru jsou umístěny tři Hallové snímače, které slouží k zjištění přesné polohy rotoru

při otáčení motoru. Na základě polohy rotoru jsou vinutí statoru napájena v definovaném pořadí. Kdykoli procházejí póly rotoru snímanou oblastí, vydávají vysoký nebo nízký signál v závislosti na pólu N nebo S. Na základě těchto tří signálů lze určit přesnou sekvenci komutace. [19]



Obrázek 15: Schématické složení bezkartáčového motoru, upraveno z [22]



Obrázek 16: Bezkartáčový motor [18]

6. Použitý hardware

6.1. Maxon motor EC 90 flat s MILE encoderem

Tento typ motoru se používá v přesných aplikacích nebo v polohovacích úlohách a díky použitému encoderu typu MILE dosahuje velmi tenkého a plochého designu. MILE (**M**axon's **I**nductive **L**ittle **E**ncoder) encoder je nejmenší indukční rotační snímač. Funguje na principu detekce vysokofrekvenční indukivity, která generuje vířivý proud v elektricky vodivém místě. Tento motor lze použít jako pohonnou jednotku např. v logistické robotice. [24]

Výhody:

- Vysoká odolnost proti prachu a oleji
- Vysoká rychlost
- Citlivost proti interferenčnímu impulsu

Vlastnosti:

- Typ motoru - Bezkartáčový
- Encoder - Ano
- Nominální napětí - 36V
- Nominální proud - 1.78A
- Rychlost - 1240 rpm
- Točivý moment - 405 mNm



Obrázek 17: Maxon motor EC 90 flat s MILE encoderem [24]

6.2. Maxon motor driver EPOS2 50/5

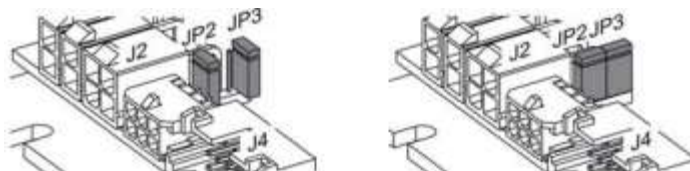
EPOS2 50/5 je malý, modulární a plně digitální polohovací regulátor, který pracuje v rozmezí 5 – 250W. Díky flexibilnímu a vysoce efektivnímu výkonovému stupni napájí stejnosměrné kartáčové motory s encoderem nebo bezkartáčové EC motory s Hallovým snímačem a snímačem polohy. Sinusová proudová komutace nabízí možnost pohánět bezkartáčové motory s minimálním krouticím momentem a s nízkou hlučností. EPOS2 50/5 je navržen k řízení, ale také, aby byl řízen v uzlu v síti CANopen, jako slave. [23]



Obrázek 18: Motor driver EPOS2 50/5 [23]

6.2.1. Konektor k motoru

Ve výchozím nastavení je regulátor nastaven k řízení stejnosměrných kartáčových nebo bezkartáčových motorů s odděleným vodičem pro encoder. Pro použití stejnosměrného motoru s integrovaným encoderem je nutné změnit polohy propojky (jumper) viz Obrázek 19. [23]



Obrázek 19: Propojky (JP2/JP3) – otevřeno (Výchozí stav) / zavřeno (Vpravo) [23]

Tabulka 6: Zapojení motoru [23]

Pin	Bezkartáčové EC motory		Kartáčové motory	
	Signál	Popis	Signál	Popis
1	Vinutí motoru 1	EC motor: Vinutí 1	Motor (+M)	DC Motor: Motor +
2	Vinutí motoru 2	EC motor: Vinutí 2	Motor (-M)	DC Motor: Motor -
3	Vinutí motoru 3	EC motor: Vinutí 3	Nepřipojen	
4	Stínění motoru	Stínění kabelu	Stínění motoru	Stínění kabelu

6.2.2. Konektor k Encoderu

Ve výchozím nastavení je regulátor nastaven pro snímání 500 otáček. U ostatních snímačů je nutné upravit nastavení pomocí softwaru.[23]

Tabulka 7: Zapojení encoderu [23]

Pin	Signál	Popis
1	EC motor: žádný DC motor: Motor +	
2	5 VDC / 100mA	Napájecí napětí
3	GND	Uzemnění
4	EC motor: žádný DC motor: Motor -	
5	Kanál A\	Kanál A doplňkový
6	Kanál A	Kanál A
7	Kanál B\	Kanál B doplňkový
8	Kanál B	Kanál B
9	Kanál I\	Doplňkový index
10	Kanál I	Index

6.2.3. Konektor k Hall senzoru

K použití Hall senzoru v bezkartáčovém motoru je nutná detekce polohy rotoru.

Tabulka 8: Zapojení Hall senzoru [23]

Pin	Signál	Popis
1	Hall senzor 1	Hall senzor 1 vstup
2	Hall senzor 2	Hall senzor 2 vstup
3	Hall senzor 3	Hall senzor 3 vstup
4	GND	Uzemnění Hall senzoru
5	+VHALL	Napájecí napětí Hall senzoru
6	Stínění Hall senzoru	Stínění kabelu

6.3. Mikrokontrolér

Požadavkem této práce byla komunikace motor driveru s mikrokontrolérem přes sběrnici CAN. Z tohoto důvodu bylo nutné zvolit náhradu za Netduino, která neobsahuje rozhraní CAN. Vhodnými kandidáty jsou mikrokontroléry od společnosti STMicroelectronics, které nabízí velké množství vývojových kitů a splňují požadavky pro řízení motoru přes sběrnici CAN. Jako vhodná náhrada Netduina je mikrokontrolér STM32F103ZE s parametry [20]:

- 72MHz ARM Cortex-M3 LQFP
- Flash: 512KB (STM32F103ZE)
- RAM: 64KB (STM32F103ZE)
- USB 2.0 rozhraní
- CAN 2.0 rozhraní
- Potenciometr (ADC vstup – 12bit)
- Display QVGA LCD-TFT
- Joystick s 5 pozicemi
- 8 LED diod and 4 tlačítka
- Napájení přes USB
- Konektor pro Debug rozhraní (ULINK2/ME 20-pin JTAG konektor)

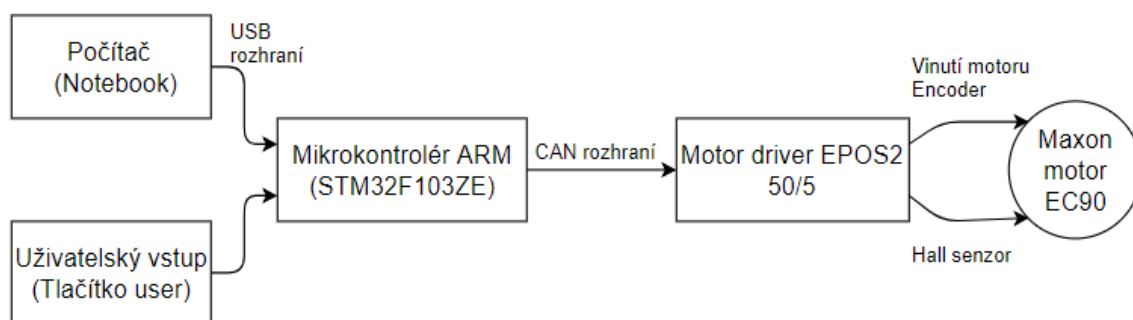


Obrázek 20: Vývojová deska STM32F103ZE [21]

6.4. Schéma zapojení

Schéma na obrázku 21 popisuje zapojení jednotlivých komponent při řízení motor driveru přes rozhraní CAN.

Počítač a uživatelský vstup tvoří vstupní blok zapojení, pomocí kterého lze ovládat mikrokontrolér. Nicméně počítač slouží hlavně k nahrávání kódu do paměti mikrokontroléru přes rozhraní USB. CAN rozhraní zajišťuje komunikaci mezi motor driverem a mikrokontrolérem pomocí příkazů, které nastavují motor driver do požadovaného stavu. Napájení a sledování polohy motoru je zajištěno pomocí motor driveru, který vyhodnocuje, zda se motor pohybuje podle nastavených hodnot.



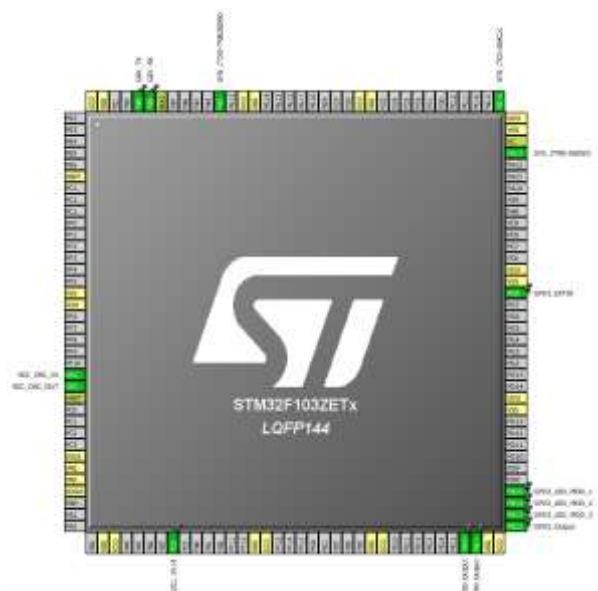
Obrázek 21: Schéma zapojení

7. Program mikrokontroléru

Pro vytvoření programu bylo použito grafické prostředí STM32CubeMX a v prostředí KEIL byl psán kód v jazyce C.

7.1. Vytvoření projektu v STM32CubeMX

Pomocí grafického rozhraní lze nakonfigurovat veškeré periferie a připravit pro IDE včetně HAL knihoven. Při vytváření projektů byl zvolen vývojový kit STM32F103ZE, po potvrzení se zobrazí vybraný mikrokontrolér s možností nastavení periferií (Obrázek 22).



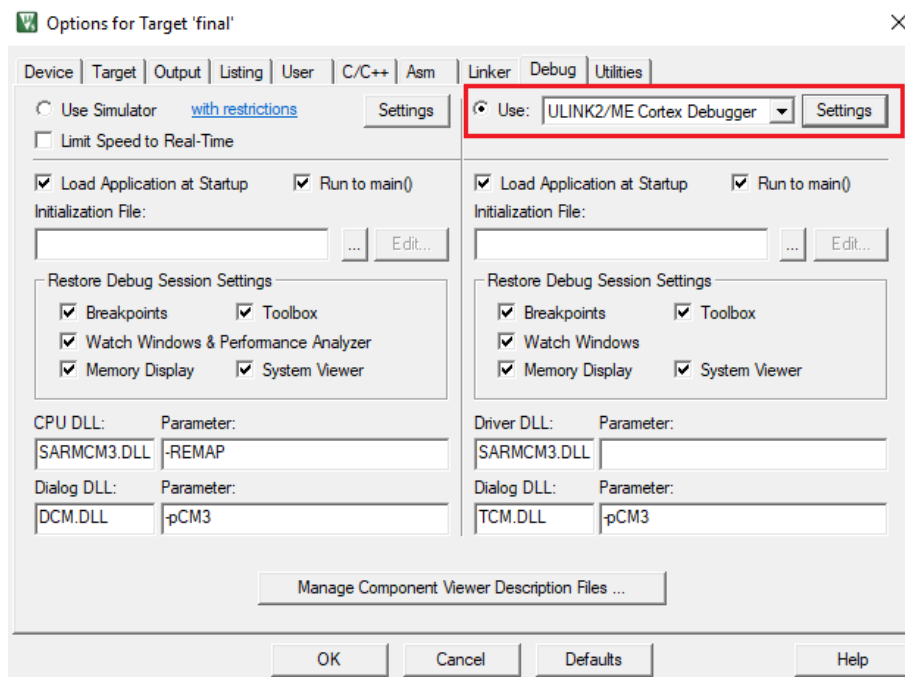
Obrázek 22: STM32CubeMX nastavení mikrokontroléru

Mezi nastavené periferie patří RCC, CAN, LED diody, tlačítko a potenciometr s ADC1.

- **Periferie RCC** – představuje generování hodinového signálu, který určuje frekvenci mikrokontroléru
- **Periferie CAN** – slouží ke komunikaci mezi jednotlivými zařízeními a využívá zmíněnou periferii RCC a diody na pinech PB8 a PB9
- **Periferie LED** – slouží pouze jako signalizační prvek funkčnosti CAN rozhraní a pro identifikaci vybraného modu. Piny PB8 – PB15
- **Tlačítko** – user tlačítko je na pinu PG8, které je vytvořeno pomocí externího přerušování, které eliminuje zákmit tlačítka (detekce vzestupné a sestupné hrany).
- **Potenciometr ADC1** – jedná se o 12-ti bitový analogově digitální převodník, který se používá k převodu analogové hodnoty z potenciometru. Pin PC4

7.2. Vývojové prostředí KEIL

Jednotlivá nastavení všech periférií lze vygenerovat pomocí zmíněného prostředí STM32CubeMX, pomocí tlačítka *Generate code* v záložce *Project*. Následně dochází ke spuštění vývojového prostředí KEIL, kde jsou vybrané periferie předpřipraveny. Pro nahrání kódu do mikrokontroléru a pro použití debuggeru je nutné nastavit v záložce *Debug* debugger s označením ULINK2/ME, viz Obrázek 23.



Obrázek 23: Keil nastavení debuggeru

7.3. Knihovna funkcí

Součástí zadání je vytvoření knihoven funkcí, které usnadňují práci při tvorbě programu. V knihovně EPOS jsou implementovány funkce pro uvedení motor driveru do aktivního / neaktivního stavu a také funkce, pomocí kterých nastavíme motor driver do požadovaného modu. K ověření funkčnosti byly vybrány mody pro nastavení polohy (Position mode) a rychlostí motoru (Profile velocity mode). Mezi vytvořené funkce knihovny EPOS patří:

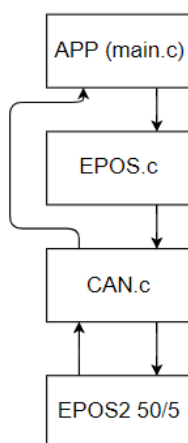
Knihovna funkcí EPOS

- **EPOS_Enable()** – slouží k uvedení motor driveru EPOS do aktivního stavu
- **EPOS_Disable()** – slouží k uvedení motor driveru EPOS do neaktivního stavu

- **EPOS_SetPositionMode()** – nastavení motor driveru do modu Position
- **EPOS_GetPositionActualValue()** – získání aktuální polohy z motor driveru
- **EPOS_SetPosition (actual_position)** – nastavení požadované polohy motoru
- **EPOS_SetVelocityMode()** - nastavení motor driveru do modu Velocity
- **EPOS_VelocityModeSetingValue (demand_velocity)** - nastavení požadované rychlosti motoru

7.4. Schéma komunikace s motor driverem

Na obrázku 24 jsou uvedeny knihovny končící příponou .c, které byly naprogramovány a nahrány do paměti STM32F103ZE. Poslední blok představuje motor driver komunikující přes CAN rozhraní pomocí knihovny CAN.c v obousměrném režimu. Z motor driveru se získávají i nastavují požadované hodnoty.

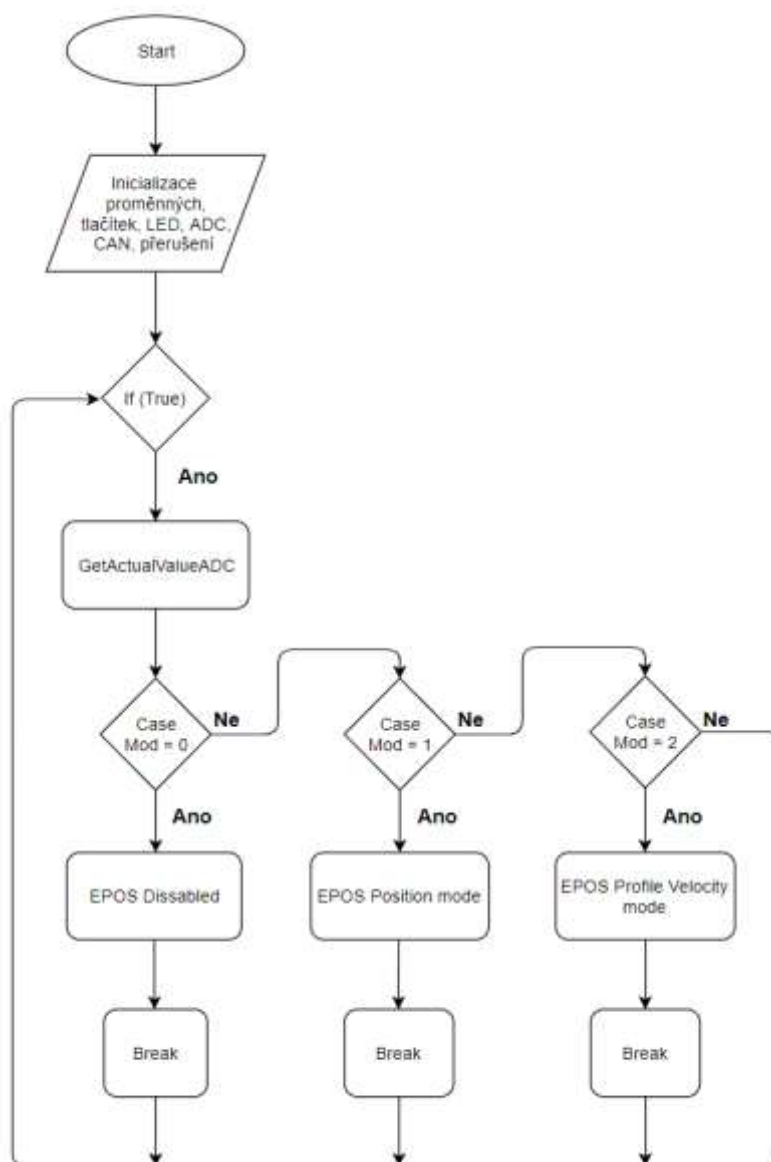


Obrázek 24: Schéma při komunikaci s EPOS2 50/5

Příkladem obousměrné komunikace motor driveru s vývojovou deskou STM32F103ZE je *Position Mode*. Z hlavního zdrojového kódu (main.c) pomocí funkce *EPOS_GetPositionActualValue()* získáváme údaje o aktuální poloze motoru. K odeslání požadavku slouží funkce *SendSDOPacket*, při které obratem dostáváme informaci o aktuální pozici motoru. Tato informace je přečtena funkcí *HAL_CAN_RxCpltCallback* z knihovny CAN.c. Poté se každou 1ms nastavuje aktuální pozice z potenciometru.

Při jednosměrné komunikaci dochází pouze k nastavování motor driveru, ať už se jedná o aktuální rychlost, nastavení pracovního režimu motor driveru nebo uvedení motor driveru do Enable/Disable režimu.

7.5. Vývojový diagram programu



Obrázek 25: Vývojový diagram programu

Na začátku programu dochází k inicializaci základních proměnných, komunikaci CAN rozhraní, LED diod, tlačítek, A/D převodníku a přerušení. Poté následuje nekonečná *while* smyčka, která se cyklicky opakuje každou 1ms. V každém cyklu je získaná hodnota z A/D převodníku, která odpovídá aktuální poloze potenciometru. V závislosti na zvoleném modu se vykonávají instrukce, které nastavují motor driver do požadovaného stavu (Disabled, Position mode a Profile velocity mode). Tento stav lze změnit pomocí tlačítka USER na vývojové desce mikrokontroléru.

Position mode porovnává skutečnou polohu od polohy požadované. Požadovaná poloha odpovídá hodnotě natočení potenciometru o definovaný úhel, je-li tato hodnota natočení větší než maximální krok, dochází k postupnému natáčení motoru, tak aby nebyla překročena maximální rychlost.

Profile velocity mode nastavuje požadovanou rychlost motoru pomocí potenciometru. Ve středu rozsahu potenciometru je tzv. dead zona, která reprezentuje nulovou rychlost motoru. Směr otáčení motoru je definován směrem opuštění dead zony a rychlost je úměrná vzdálenosti od dead zony. Mód s označením profile zajistí nepřekročení maximální rychlosti.

Závěr

Bakalářská práce popisuje řízení motor driveru pomocí mikrokontroléru třídy ARM prostřednictvím CAN rozhraní. V teoretické části práce jsou popsány dvě rodiny motor driverů od společností Maxon a Roboteq. Zadání práce se mírně upravilo, z důvodu nepřítomnosti CAN rozhraní na vývojové desce Netduina, což se vyřešilo změnou za STM32F103ZE.

V praktické části byla realizována kabeláž mezi jednotlivými komponenty. Po softwarové stránce se nakonfigurovala vývojová deska STM32F103ZE prostřednictvím STM32CubeMX. V programovém prostředí KEIL jsou vytvořeny knihovny v jazyce C, které byly využity k otestování funkcí.

Seznam použité literatury

- [1] *Motor Drivers vs. Motor Controllers [online]. [cit. 2018-05-13]. DOI: <https://core-electronics.com.au/tutorials/motor-drivers-vs-motor-controllers.html>.*
- [2] *Motor Drivers [online]. [cit. 2018-05-13]. Dostupné z: <http://www.futureelectronics.com/en/drivers/motor-driver.aspx>.*
- [3] *UZIMEX – firemní stránky Maxon řízení EPOS [online]. [cit. 2018-05-13]. Dostupné z: http://www.uzimex.cz/soubory/20060628_tt_2006-serial-epos.pdf.*
- [4] *EPOS2 Positioning Controllers [online]. [cit. 2018-05-13]. Dostupné z: https://www.maxonmotor.nl/medias/sys_master/root/8827037122590/17-EN-431-432-433-435.pdf.*
- [5] *UZIMEX – FIREMNÍ STRÁNKY. Maxon Motor: Jednotky řízení polohy EPOS [online]. [cit. 2018-05-13]. Dostupné z: http://www.uzimex.cz/soubory/20050608_275512_cz_pruvodce.pdf.*
- [6] *Maxon Motor: EPOS2 50/5 Firmware Specification [online]. In: . 2013, s. 248 [cit. 2018-05-21]. Dostupné z: https://www.maxonmotor.es/medias/sys_master/8811528749086/EPOS2-Firmware-Specification-En.pdf.*
- [7] *RoboteQ Motor Controller [online]. [cit. 2018-05-13]. Dostupné z: http://www.roborealm.com/help/RoboteQ_Motor_Control.php.*
- [8] *RoboteQ - XDC2460 [online]. [cit. 2018-05-13]. Dostupné z: <https://www.roboteq.com/index.php/roboteq-products-and-services/brushed-dc-motor-controllers/326/xdc2230-319-detail>.*
- [9] *RoboteQ - Brushed DC Motor Controllers [online]. [cit. 2018-05-13]. Dostupné z: <https://www.roboteq.com/index.php/roboteq-products-and-services/brushed-dc-motor-controllers#ProdTables>.*
- [10] *Brushless DC Motor Controllers [online]. [cit. 2018-05-13]. Dostupné z: <https://www.roboteq.com/index.php/roboteq-products-and-services/brushless-dc-motor-controllers#ProdTables>.*
- [11] *Microcontrollers [online]. [cit. 2018-05-13]. Dostupné z: <http://www.futureelectronics.com/en/Microcontrollers/microcontrollers.aspx>.*
- [12] *About Netduino [online]. [cit. 2018-05-13]. Dostupné z: <http://developer.wildernesslabs.co/Netduino/About/>.*
- [13] *ARM. ARM Architecture Reference Manual [online]. Copyright © 1996- 1998, 2000, 2004, 2005 ARM Limited, Issue I, July 2005 [cit 2018-05-13]. Dostupné z: https://www.scss.tcd.ie/John.Waldron/3d1/arm_arm.pdf*

-
- [14] ZELTWANGER, Holger. *CANopen – vyšší komunikační protokol pro vestavné sítě* [online]. [cit. 2018-05-13]. Dostupné z: http://automa.cz/cz/casopis-clanky/canopen-vyssi-komunikacni-protokol-pro-vestavne-site-2004_04_32279_854/
- [15] *Řízení pohonů pomocí PLC s využitím sběrnice CAN* [online]. BRNO, 2008 [cit. 2018-05-13]. Dostupné z: https://www.vutbr.cz/www_base/zav_prace_soubor_verejne.php?file_id=8973
- [16] *Brushed DC Motors Vs. Brushless DC Motors* [online]. 01/24/2017. [cit. 2018-05-13]. Dostupné z: <https://www.motioncontrolonline.org/blog-article.cfm/Brushed-DC-Motors-Vs-Brushless-DC-Motors/24>
- [17] CONDIT, Reston. *Brushed DC Motor Fundamentals*. Microchip Technology Inc. [online]. 2004 [cit. 2018-05-13]. Dostupné z: <http://ww1.microchip.com/downloads/en/appnotes/00905a.pdf>.
- [18] *Power Tool Tech: Brushless Motors 101*. Dostupné z: <http://toolguyd.com/power-tool-brushless-motors/>
- [19] YEDAMALE, Padmaraja. *Power Tool Tech: Brushless Motors 101*. Microchip Technology Inc. 2003. Dostupné z: <http://ww1.microchip.com/downloads/en/AppNotes/00885a.pdf>.
- [20] ARM Keil: *MCBSTM32EXL* [online]. [cit. 2018-05-13]. Dostupné z: <http://www.keil.com/arm/mcbstm32exl/>
- [21] ARM Keil: *MCBSTM32E Quick Start Guide* [online]. , 3 [cit. 2018-05-13]. Dostupné z: http://www.mouser.com/ds/2/211/mcbstm32e_quickstart-15531.pdf
- [22] *Brushed vs Brushless Motors* [online]. [cit. 2018-05-13]. Dostupné z: <http://www.thinkrc.com/faq/brushless-motors.php>
- [23] *Maxon Motor: EPOS2 50/5 Hardware reference* [online]. In: . 2014, s. 50 [cit. 2018-05-21]. Dostupné z: https://www.maxonmotor.com/medias/sys_master/8815046295582/347717-Hardware-Reference-En.pdf
- [24] *Maxon Motor: MILE Encoder for EC 90 flat - Product Information* [online]. In: . 2014, s. 10 [cit. 2018-05-21]. Dostupné z: https://www.maxonmotor.com/medias/sys_master/root/8814678704158/enx-ec90flat-mile-okt-en.pdf
- [25] *RoboteQ: Motor Controller Modes* [online]. [cit. 2018-05-21]. Dostupné z: <https://www.roboteq.com/index.php/technology-menu/421-motor-controller-modes>
-